(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 January 2001 (11.01.2001)

PCT

(10) International Publication Number
WO 01/02954 A1

(51) International Patent Classification⁷: G06F 9/445, 9/44

(21) International Application Number: PCT/EP00/05952

(22) International Filing Date: 27 June 2000 (27.06.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/343,607 30 June 1999 (30.06.1999) US

(71) Applicant: KONINKLIJKE PHILIPS ELECTRONICS N.V. [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(72) Inventors: ALSAFADI, Yasser; Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL). SCHAFFER, James, D.; Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

(74) Agent: GRAVENDEEL, Cornelis; Internationaal Octrooibureau B.V., Prof Holstlaan 6, NL-5656 AA Eindhoven (NL).

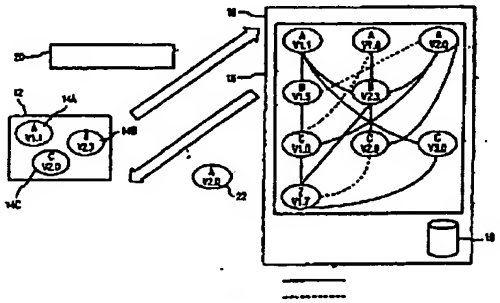
(81) Designated States *(national)*: JP, KR.

(84) Designated States *(regional)*: European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published:
— *With international search report.*
— *Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: RECONFIGURATION MANAGER FOR CONTROLLING UPGRADES OF ELECTRONIC DEVICES



(57) Abstract: A reconfiguration manager implemented on a computer or other data processing device controls the reconfiguration of software or other components of an electronic device such as a computer, personal digital assistant (PDA), set-top box, television, etc. The reconfiguration manager receives a reconfiguration request, e.g., a software upgrade request from the electronic device, and determines one or more device components that are required to implement the reconfiguration request. The reconfiguration manager also determines, e.g., from information in the request, identifiers of one or more additional components currently implemented in the electronic device. The reconfiguration manager then compares the needed and currently implemented components with previously-stored lists of known acceptable and unacceptable configurations for the electronic device. If the needed and currently implemented components correspond to a configuration on the list of acceptable configurations, the request is approved and the needed components are downloaded to the electronic device. If the needed and currently implemented components correspond to a configuration on the list of unacceptable configurations, the request is denied. Otherwise, the reconfiguration manager may indicate that the requested reconfiguration is unknown, or may take another action such as responding to the electronic device with a list of other components that would be required to implement the request.

WO 01/02954 A1

Reconfiguration manager for controlling upgrades of electronic devices

Field of the Invention

The present invention relates generally to the field of electronic devices, and more particularly to techniques for upgrading or otherwise reconfiguring software and/or hardware components in such devices.

Background of the Invention

For many different electronic devices, such as desktop, laptop and palmtop computers, personal digital assistants (PDAs), telephones, televisions, set-top boxes and other consumer electronic processing devices, it is common for ongoing development efforts to continue to produce improvements to existing device software or hardware components, as well as new components that add to or otherwise improve device functionality. Users of such devices often prefer to upgrade their devices incrementally, rather than discard their current devices and purchase new ones. However, for most contemplated upgrades, it is generally necessary to determine if the new or improved component is compatible with the rest of the device, and if not, what other components would need simultaneous upgrading in order to provide the desired compatibility. This compatibility determination can be particularly difficult if the range of possible device configurations is large and the interaction among device components is complex.

A number of different techniques have been developed for updating components of electronic devices. For example, U.S. Patent No. 5,155,847 discloses a technique for updating software at remote locations. A central computer system stores the original software, and keeps track of all the software configurations for a number of remote systems. The remote system software is upgraded or otherwise changed based on patches transmitted by the central computer system. However, this technique generally requires the central computer system to keep track of the particular software configurations at each of the remote systems. Furthermore, the technique is not directly applicable to electronic devices other than computers, and cannot efficiently handle reconfiguration of hardware components, or hardware and software interdependencies.

Another conventional technique, described in PCT Application No. WO 94/25923, manages the configuration of an enterprise-wide network which includes at least one centralized computer and a plurality of desktop computers. The technique attempts to ensure that each of the desktop computers has an appropriate set of resources as determined in accordance with a set of enterprise policies. However, the technique generally assumes that the resources required by each desktop computer are independent, and fails to adequately address situations in which the required resources are highly interdependent. Furthermore, this technique generally assumes that the information regarding component interactions is fully specified and built in to the system.

UK Patent Application No. GB 2,325,766 discloses a version management system for keeping files on remote devices updated to latest versions as determined by a master list maintained on a central server. The updating process in this approach generally involves adding, amending and deleting files in their entirety. A significant problem with this approach is that it apparently assumes either that the files are independent or that any potential conflicting requirements have already been resolved using other techniques. It fails to provide generalized techniques for ensuring compatibility among requested components.

A convention technique disclosed in PCT Application No. WO 96/32679 describes the remote patching of operating code in a mobile unit of a distributed system. A manager host device in the system transmits patches to the mobile unit, and the mobile unit creates patched operating code by merging the patches with current operating code and switching execution to the patched operating code. However, like the other conventional techniques described previously, this technique also fails to adequately ensure compatibility among software and hardware components for a variety of different electronic devices.

As is apparent from the above, a need exists for improved techniques for managing reconfiguration of electronic devices, such that compatibility determinations can be facilitated, particularly for large and complex device configurations.

Summary of the Invention

The invention provides a reconfiguration manager that may be implemented on a computer or other data processing device to control the reconfiguration of software or other components of an electronic device such as a computer, personal digital assistant (PDA), set-top box, television, etc. In accordance with the invention, a reconfiguration manager receives a reconfiguration request, e.g., a software upgrade request from the electronic device, and determines one or more device components that are required to

implement the reconfiguration request. The reconfiguration request can be received directly from the electronic device itself, or otherwise supplied to the reconfiguration manager.

The reconfiguration manager also determines, e.g., from information supplied by the electronic device as part of the request, identifiers of one or more additional components currently implemented in the electronic device. The reconfiguration manager then compares the needed and currently implemented components with previously-stored lists of known acceptable and unacceptable configurations for the electronic device. If the needed and currently implemented components correspond to a configuration on the list of acceptable configurations, the request is approved and the needed components are downloaded or otherwise supplied to the electronic device. If the needed and currently implemented components correspond to a configuration on the list of unacceptable configurations, the request is denied. Otherwise, the reconfiguration manager may indicate that the requested reconfiguration is unknown, or may take another action such as responding to the electronic device with a list of other components that would be required to implement the reconfiguration request.

Advantageously, the invention provides efficient techniques for incrementally upgrading or otherwise reconfiguring electronic devices. The invention ensures that upgrades are compatible with the configuration of a given device before they are implemented in that device, thereby avoiding problems associated with inconsistent upgrades. Although particularly well suited for use with software upgrades delivered over a network, the invention is applicable to reconfiguration of other types of device components, e.g., hardware components or combinations of hardware and software components, and to numerous other applications. These and other features and advantages of the present invention will become more apparent from the accompanying drawings and the following detailed description.

Brief Description of the Drawings

FIG. 1 illustrates the operation of a reconfiguration manager in accordance with a preferred embodiment of the invention.

FIG. 2 is a flow diagram showing processing operations implemented in the reconfiguration manager of FIG. 1.

FIG. 3 is a block diagram of an exemplary network-based computer system which includes a reconfiguration manager in accordance with the invention.

Detailed Description of the Invention

FIG. 1 shows a preferred embodiment of the invention, in which a reconfiguration manager 10 interacts with an electronic device 12 also referred to as "Device". The device 12 may represent a desktop, laptop or palmtop computer, a personal digital assistant (PDA), a telephone, television, set-top box or any other type of consumer electronic processing device. The device 12 includes a number of software components 14A, 14B and 14C, corresponding to version 1.1 of a software component A, version 2.3 of a software component B, and version 2.0 of a software component C, respectively. The reconfiguration manager 10 may be implemented on a computer, a set of computers, or any other type of data processing system or device.

The reconfiguration manager 10 includes a listing 16 of known configurations, and a repository 18 of software components. Repository 18 may represent, e.g., a database, data warehouse, physical warehouse or any other type of storage device or element incorporated in or otherwise associated with a computer or other processing system or device on which the reconfiguration manager 10 is implemented. The repository 18 need not be co-located with the processing portions of the reconfiguration manager 10. For example, the repository 18 could be accessed by the reconfiguration manager 10 over a suitable network connection.

The list 16 in this example is illustrated in the form of a graph indicating which of a set of software components supported by the manager 10 are known to work well together or are otherwise compatible. The list 16 includes identifiers of a number of software components, each represented by an oval, including components corresponding to versions 1.1, 1.8 and 2.0 of the software component A, versions 1.5 and 2.3 of the software component B, versions 1.0, 2.0 and 3.0 of a software component C, and version 1.7 of a software component Z. Each of at least a subset of these components of the list 16 may be stored in the software component repository 18. Additional components not shown may also be stored in the repository 18.

A solid line between a given pair of components in the exemplary list 16 indicates that the pair of components corresponds to a known "good" configuration, i.e., the components work well together or are otherwise compatible. The pair including version 1.1 of component A and version 1.5 of component B is an example of a known good configuration. A dashed line between a given pair of components in the list 16 indicates that the pair of components correspond to a known Abad≅ configuration, i.e., are not compatible.

The pair including version 1.8 of component A and version 1.0 of component C is an example of a known bad configuration.

It should be understood that the list 16, although shown in graphical form in FIG. 1, may be implemented, e.g., as a stored table, set of tables or other type of list in a memory of the reconstruction manager 10, as a portion of a program executed by the reconfiguration manager 10, or in any other suitable format. Moreover, although illustrated in FIG. 1 as indicating pair-wise compatibility among components, the list in other embodiments could include information indicative of compatibility between groups of multiple components. The term "list" as used herein is therefore intended to include any stored representation of information indicative of component compatibility. A given stored list in accordance with the invention can be implemented in a straightforward manner, as will be apparent to those skilled in the art.

In operation, the reconfiguration manager 10 receives a request 20 from the device 12. In this example, the request 20 indicates that a user of the device 12 wants to upgrade the device to include version 2.0 of software component A. The request in the illustrative embodiment also includes a list of the components currently in the device, i.e., version 1.1 of component A, version 2.0 of component C and version 2.3 of component B. The request may include additional information, such as any needed information regarding the interconnection of the components or other parameters associated with the device. The reconfiguration manager 10 processes the request, in a manner to be described in greater detail in conjunction with the flow diagram of FIG. 2, and if appropriate delivers to device X a response 22 which includes the requested version 2.0 of software component A.

For example, the reconfiguration manager first determines whether the requested upgrade, in this case version 2.0 of component A, is compatible with other components of device X, i.e., version 2.3 of component B and version 2.0 of component C. The reconfiguration manager 10 in the embodiment of FIG. 1 makes this determination using the list 16. In this case, list 16 indicates that version 2.0 of component A is compatible with version 2.3 of component B and version 2.0 of component C. As a result, the requested upgrade is delivered to device 12 as part of the response 22.

FIG. 2 shows a flow diagram illustrating the operation of the reconfiguration manager 10 in greater detail. In step 100, the reconfiguration manager 10 obtains information regarding the hardware and software configuration of device X, i.e., electronic device 12 of FIG. 1. This information is generally included as part of the request 20 sent by the device 12 to the reconfiguration manager 10. In other embodiments, this information may be obtained

in another suitable manner, e.g., from a local database based on a serial number or other identifier of the electronic device.

In step 102, the reconfiguration manager 10 determines that the request 20 includes a request for a software upgrade, i.e., a request to upgrade to version 2.0 of component A. It should be noted that, although described primarily in conjunction with software upgrades, the invention is also applicable to hardware upgrades, and to upgrades in combinations of hardware and software, as well as to other changes in device configuration. In the FIG. 2 example, the request is for an upgrade to a particular software component. Other types of requests which may be processed by the reconfiguration manager 10 of FIG. 1 include requests for an upgrade to a particular device feature. Such a feature upgrade may require the reconfiguration manager to upgrade several device components.

In step 104 of FIG. 2, the reconfiguration manager 10 generates a potential upgrade configuration that will satisfy the received request. The reconfiguration manager in step 106 then searches through a set of known bad configurations. If the upgrade configuration as generated in step 104 is determined in step 108 to correspond to one of the known bad configurations, the reconfiguration manager in step 110 attempts to find a set or sets of potential upgrade configurations from a set of known good configurations.

If the resulting set of potential upgrade configurations is determined in step 112 to be empty, the reconfiguration manager in step 114 denies the upgrade, since it is known to be incompatible with the current configuration of device X, and communicates this denial in its response to device X. If step 112 indicates that the set is not empty, a particular set of upgrade configuration is selected in step 116, and the upgrade is approved in step 118 as compatible with the current configuration of device X. The selection in step 116 may be based at least in part on one or more established criteria, such as least expensive, maximum improvement in system operating speed, most recently modified, most energy efficient, or other suitable criteria. The reconfiguration manager or other server associated therewith then downloads the upgrade to device X in step 120.

If step 108 determines that the upgrade configuration as generated in step 104 does not correspond to a known bad configuration, the reconfiguration manager in step 122 searches the list of known good configurations to determine if the upgrade configuration determined in step 104 is a known good configuration. If it is determined in step 124 to be a known good configuration, the upgrade is approved in step 118, and the reconfiguration manager or other server associated therewith downloads the upgrade to device X in step 120. If the configuration is not a known good configuration, the reconfiguration manager in step

130 returns in its response to the device X an indication that the requested upgrade is "fuzzy" or unknown, e.g., not known to be valid.

Other types of responses that may be generated by the reconfiguration manager 10 include, e.g., a response which includes a list of additional components that are prerequisites for the requested upgrade. This type of response may provide a user associated with device X with an option to download all of the components required to implement the desired upgrade.

FIG. 3 shows an example of a system 200 in which a reconfiguration manager in accordance with the invention may be implemented. The system 200 includes reconfiguration manager 10 and electronic device 12 as previously described in conjunction with FIGS. 1 and 2. The reconfiguration manager 10 and electronic device 12 are connected with a number of server devices 210 and client devices 212 over a network 214. As previously noted, the reconfiguration manager 10 and electronic device 12 may be implemented as computers or other electronic data processing devices. In this example, the electronic device 12 includes a processor 220 and a memory 222, and the reconfiguration manager 10 includes a processor 230 and a memory 232.

The processors 220 and 230 may represent, e.g., microprocessors, central processing units, computers, circuit cards, application-specific integrated circuits (ASICs), as well as portions or combinations of these and other types of processing devices. The memories 222 and 232 may represent, e.g., disk-based optical or magnetic storage units, electronic memories, as well as portions or combinations of these and other memory devices.

The functional operations associated with the reconfiguration manager 10 and electronic device 12, as described in detail in conjunction with FIGS. 1 and 2, may be implemented in whole or in part in one or more software programs stored in their respective memories 222, 232 and executed by their respective processors 220, 230. The network 214 may represent a global computer communications network such as the Internet, a wide area network, a metropolitan area network, a local area network, a cable network, a satellite network or a telephone network, as well as portions or combinations of these and other types of networks. Reconfiguration manager 10 and device 12 may themselves be respective server and client machines coupled to the network 214.

It should be noted that the reconfiguration manager need not receive a reconfiguration request directly from the electronic device itself. For example, it is possible for the reconfiguration manager to receive requests from an intermediary, e.g., a server or other designated machine which collects reconfiguration requests from multiple devices or

users and delivers the requests in an appropriate manner to the reconfiguration manager. As another example, a help desk operator or other human or machine interface can receive reconfiguration requests from users of electronic devices. In such applications, information identifying the electronic device, e.g., the device serial number, may be supplied by the user. Information regarding the particular components in the device may be determined, e.g., by accessing a local database using the device identifying information, may be supplied directly by the user, or may be determined using combinations of these and other techniques.

The above-described embodiments of the invention are intended to be illustrative only. For example, the invention can be used to implement upgrading or other reconfiguration of any desired type of software or hardware component, as well as combinations of these and other components, for any desired type of electronic device, and in many applications other than those described herein. The invention can also be implemented at least in part in the form of one or more software programs which are stored on an otherwise conventional electronic, magnetic or optical storage medium and executed by a processing device, e.g., by the processors 220 and 230 of system 200. These and numerous other embodiments within the scope of the following claims will be apparent to those skilled in the art.

CLAIMS:

1. A processor-implemented method for controlling the reconfiguration of an electronic device (12), the method comprising the steps of:

receiving information representative of a reconfiguration request (20) relating to the electronic device;

determining at least one device component required to implement the reconfiguration request;

comparing the determined component and information specifying at least one additional component (14A, 14B, 14C) currently implemented in the electronic device with at least one of a list (16) of known acceptable configurations for the electronic device and a list (16) of known unacceptable configurations for the electronic device; and

generating information (22) indicative of an approval or a denial of the reconfiguration request based at least in part on the result of the comparing step.

2. An apparatus for controlling the reconfiguration of an electronic device (12), the apparatus comprising:

a memory (232) for storing at least one of a list (16) of known acceptable configurations for the electronic device and a list (16) of known unacceptable configurations for the electronic device; and

a processor (230) coupled to the memory and operative (i) to receive information representative of a reconfiguration request (20) relating to the electronic device; (ii) to determine at least one device component required to implement the reconfiguration request; (iii) to compare the determined component and information specifying at least one additional component (14A, 14B, 14C) currently implemented in the electronic device with at least one of the list of known acceptable configurations for the electronic device and the list of known unacceptable configurations for the electronic device; and (iv) to generate information (22) indicative of an approval or a denial of the reconfiguration request based at least in part on the comparison operation.

3. The apparatus of claim 2 wherein the processor is further operative to generate information indicative of an approval of the reconfiguration request if the determined component and the additional component are consistent with a given one of the known acceptable configurations.

4. The apparatus of claim 2 wherein the processor is further operative to download the determined component to the electronic device if the determined component and the additional component are consistent with a given one of the known acceptable configurations.

5. The apparatus of claim 4 wherein the processor is further operative to compare the determined component and information specifying at least one additional component currently implemented in the electronic device with the list of known unacceptable configurations for the electronic device; and to generate information indicative of a denial of the reconfiguration request if the determined component and the additional component are consistent with a given one of the known unacceptable configurations.

6. The apparatus of claim 2 wherein the processor is further operative to compare the determined component and information specifying at least one additional component currently implemented in the electronic device with a list of known unacceptable configurations for the electronic device; and to generate information indicating that the requested reconfiguration is unknown if the determined component and the additional component are not consistent with a given one of the known acceptable or unacceptable configurations.

7. The apparatus of claim 2 wherein the processor is further operative to transmit in response to the reconfiguration request a list of additional components required in the electronic device in order to implement the reconfiguration request.

8. The apparatus of claim 2 wherein the information specifying at least one additional component currently implemented in the electronic device includes identifiers of each of the components in a set of components currently implemented in the electronic device.

9. The apparatus of claim 2 wherein the identifiers of each of the components in the set of components are included in the reconfiguration request transmitted by the electronic device.

10. The apparatus of claim 2 wherein the reconfiguration request comprises a request for an upgrade of at least one of a software component and a hardware component of the electronic device.

11. The apparatus of claim 2 wherein the reconfiguration request is received from the electronic device over a network connection established with a reconfiguration manager (10) which includes the memory and processor.

12. An article of manufacture comprising a machine-readable medium containing one or more software programs which when executed implement the steps of:

receiving information representative of a reconfiguration request (20) relating to an electronic device (12);

determining at least one device component required to implement the reconfiguration request;

comparing the determined component and information specifying at least one additional component (14A, 14B, 14C) currently implemented in the electronic device with at least one of a list (16) of known acceptable configurations for the electronic device and a list (16) of known unacceptable configurations for the electronic device; and

generating information (22) indicative of an approval or a denial of the reconfiguration request based at least in part on the result of the comparing step.

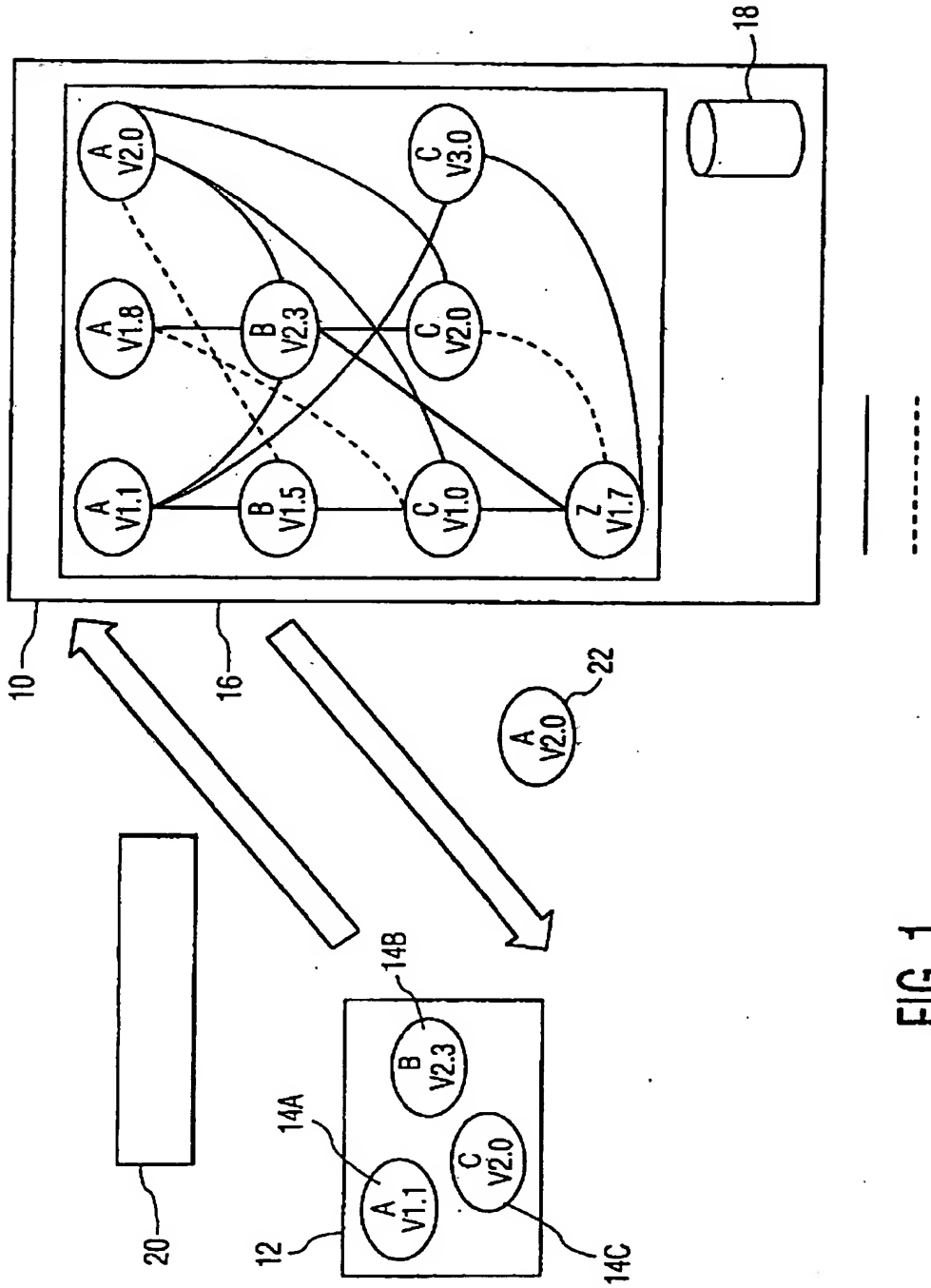

18
A V2.0
C V3.0
A V1.8
B V2.3
C V2.0
A V1.1
B V1.5
C V1.0
Z V1.7
10
16
22
A V2.0
14B
B V2.3
14A
C V2.0
A V1.1
20
12
14C

FIG. 1

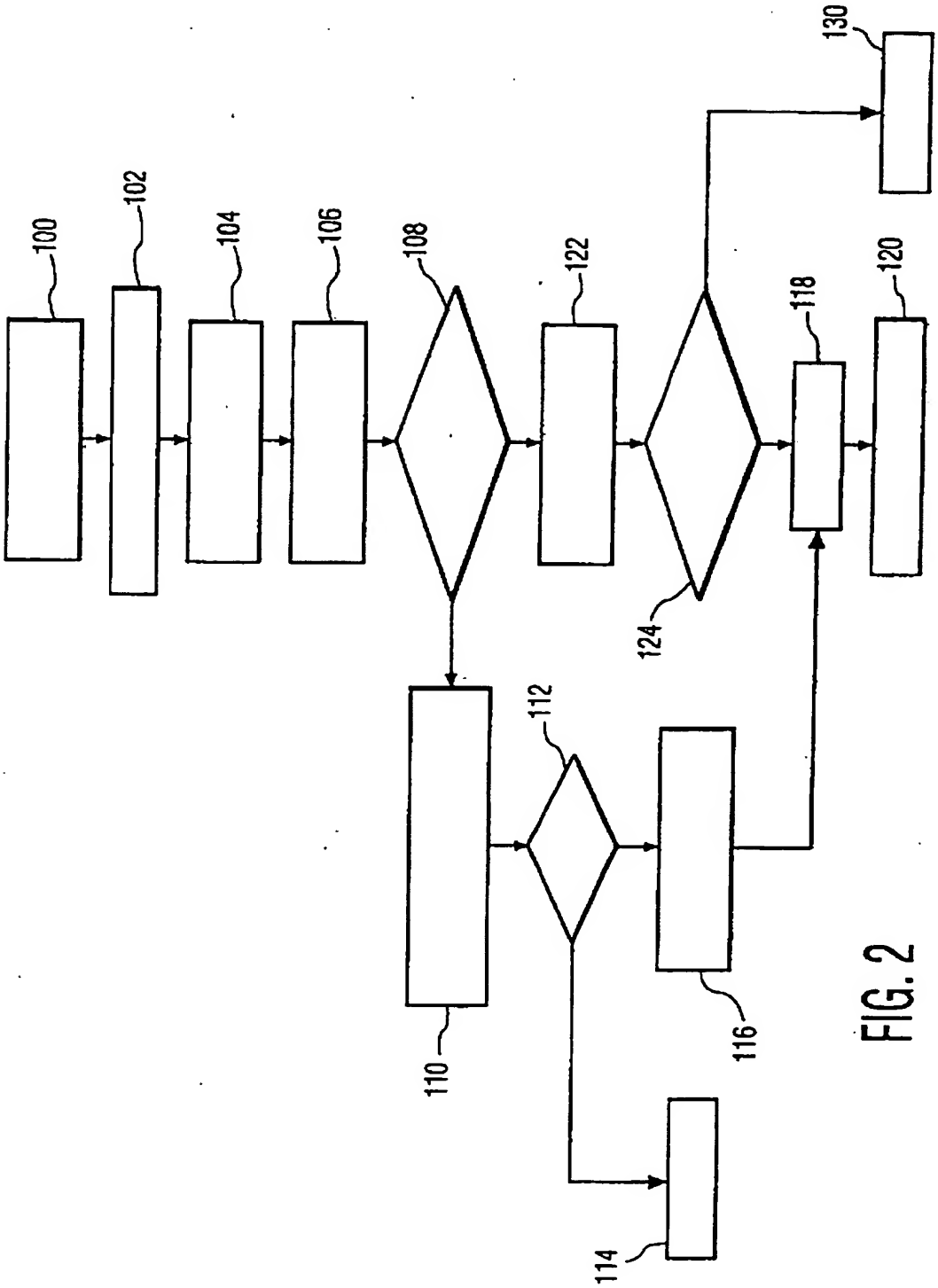


FIG. 2

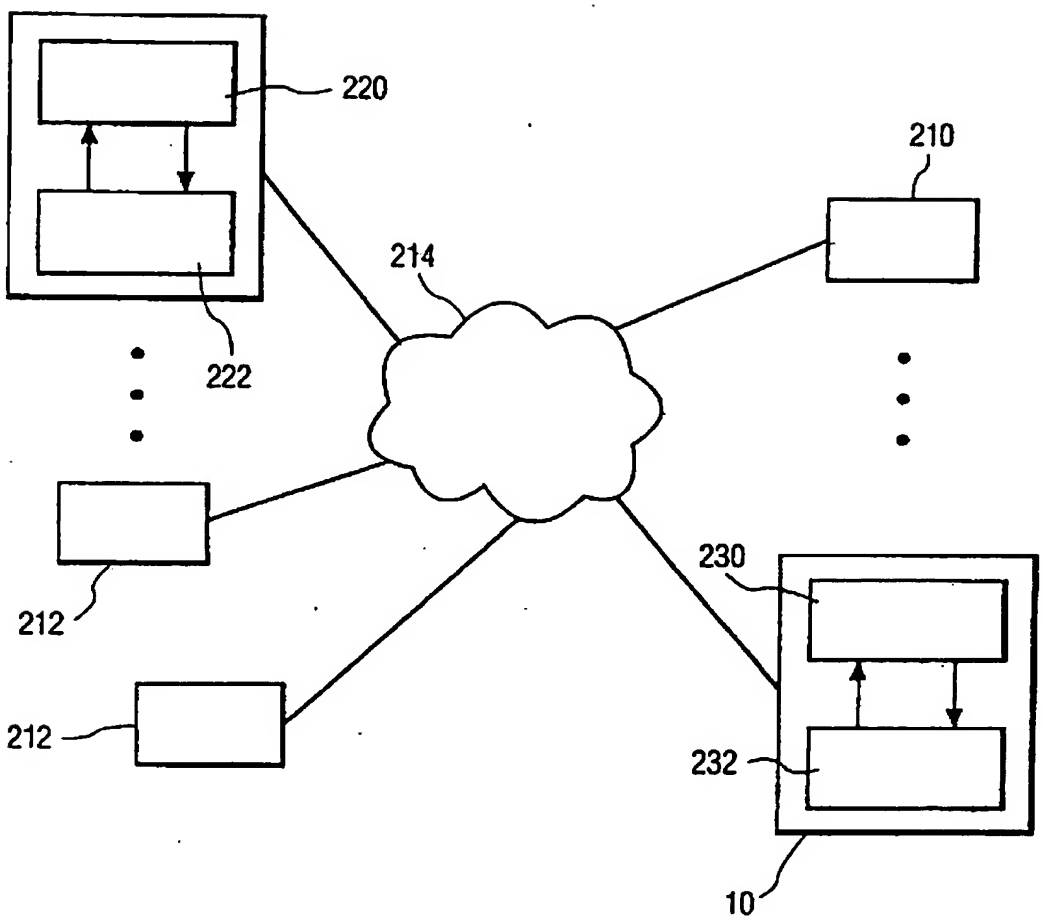


FIG. 3

INTERNATIONAL SEARCH REPORT

Inte onal Application No
PCT/EP 00/05952

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/445 G06F9/44

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, INSPEC, IBM-TDB, COMPENDEX

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 809 287 A (SHAFFER DAVID SCOTT ET AL) 15 September 1998 (1998-09-15) column 1, line 60 -column 2, line 57 column 3, line 22 - line 30 column 4, line 16 - line 24 column 6, line 36 - line 42	1,2,7,8, 11,12
A	US 5 784 702 A (GREENSTEIN PAUL GREGORY ET AL) 21 July 1998 (1998-07-21) abstract; figures 1,16 column 2, line 36 - line 59 column 3, line 59 - line 66 column 5, line 5 - line 7	1-3,8,12

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
31 October 2000	07/11/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Carciofi, A

2

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/EP 00/05952

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication,where appropdate, of the relevant passages	Relevant to claim No.
A	US 5 499 357 A (SONTY ATASHI C ET AL) 12 March 1996 (1996-03-12) abstract column 5, line 60 -column 6, line 16 column 7, line 3 - line 21	1,2,8, 10,12

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

PAGE 39/40 * RCVD AT 6/17/2005 5:05:23 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/3 * DNIS:8729306 * CSID:+12123195101 * DURATION (mm-ss):17-04

INTERNATIONAL SEARCH REPORT

information on patent family members

Inter. nal Application No
PCT/EP 00/05952

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5809287 A	15-09-1998	US 5586304 A	17-12-1996
		US 5588143 A	24-12-1996
		US 5960189 A	28-09-1999
		AU 695638 B	20-08-1998
		AU 3053895 A	21-03-1996
		CA 2157728 A,C	09-03-1996
		EP 0703531 A	27-03-1996
		JP 8227355 A	03-09-1996
US 5784702 A	21-07-1998	US 5659786 A	19-08-1997
		CA 2100540 A	20-04-1994
		EP 0593874 A	27-04-1994
		JP 7295841 A	10-11-1995
US 5499357 A	12-03-1996	BR 9402027 A	13-12-1994
		EP 0632371 A	04-01-1995
		JP 7006026 A	10-01-1995

Form PCT/ISA/210 (patent family annex) (July 1992)